IN THE CLAIMS:

Please amend the claims as follows.

---

1. (Currently Amended) A method of downloading software in native code to a receiver/decoder, comprising the steps, at the receiver/decoder of:

    receiving a bitstream including the software in native code;

    downloading into the receiver/decoder a loader in native code for loading the software from the bitstream using a bootstrap loader;

    downloading the software in native code into the receiver/decoder from the bitstream using said ~~downloaded~~ loader ~~in native code~~, and

    storing said software in native code into the receiver/decoder.

2. (Currently Amended) The method according to Claim 1, wherein the ~~downloaded data~~ loader is deleted from the receiver/decoder after the software has been downloaded from the bitstream.

3. (Currently Amended) The method according to Claim 1, wherein the ~~downloaded~~ loader is subsequently stored in non-volatile memory of the receiver/decoder.

4. (Previously Presented) The method according to Claim 3, wherein the non-volatile memory is a Flash memory volume of the receiver/decoder.

5. (Cancelled)

6. (Currently Amended) The method according to claim 1, wherein a portion only of the software stored in the receiver/decoder is replaced by a corresponding portion of the software downloaded by the ~~downloaded~~ loader.

7. - 8. (Cancelled)

2

9. (Previously Presented) The method according to Claim 44, further comprising downloading module tables having the same TID.

10.-12. (Cancelled)

13. (Previously Presented) The method according to Claim 47, wherein the version identification comprises a code for the version of the receiver/decoder and a code for the manufacturer of the receiver/decoder.

14.-19. (Cancelled)

20. (Currently Amended) A receiver/decoder comprising:

   a receiver for receiving a bitstream including software and a loader in native
      code;

   a storage means configured to store the software and the loader; and

   a downloading means for downloading configured to download the loader into the
      storage means a loader in native code for loading the software in native
      code from the bitstream[[.]],

   wherein the receiver/decoder is configured to execute the loader, and

   wherein the loader is configured to download the software into the storage means.

21. (Currently Amended) The receiver/decoder according to Claim 20, further comprising means for deleting the downloaded loader from the storage means after the software has been downloaded from the bitstream.

22. (Previously Presented) The receiver/decoder according to Claim 20, further comprising a non-volatile memory for storing the downloaded loader after the software has been downloaded from the bitstream.

23. (Previously Presented) The receiver/decoder according to Claim 22, wherein the non-volatile memory is a Flash memory volume of the receiver/decoder.

3

24. (Cancelled)

25. (Previously Presented) The receiver/decoder according to claim 20, wherein the downloaded loader is adapted to replace a portion only of the software stored in the receiver/decoder by a corresponding portion of the software downloaded thereby.

26. (Previously Presented) The receiver/decoder according to claim 20, arranged to download tables.

27. (Previously Presented) The receiver/decoder according to Claim 26, wherein said downloading means is arranged to download a table having a table identification ("TID") and a predetermined table identification extension ("TID-extension") so as to download a directory table, to determine from the content of the directory table the TID-extensions of module tables having the same TID as the directory table, and to download the module tables having the same TID as that of the downloaded directory table and TID-extensions determined from the downloaded directory table so as to download said loader.

28. (Previously Presented) The receiver/decoder according to Claim 26, wherein said downloading means is arranged to download a directory table having a predetermined TID and containing, for each of a plurality of version identifications of a receiver/decoder, a respective TID associated with that version identification, to determine the version identification of the receiver/decoder, and to download a directory table having a TID associated with a version number of the receiver/decoder and a predetermined TID-extension.

29. (Previously Presented) The receiver/decoder according to Claim 27, wherein said downloading means is arranged to determine whether a directory version identification of a currently transmitted directory table is more recent than the directory version identification of a previously downloaded directory table having the same TID as the currently transmitted directory table, and if not, to abort the downloading of said loader.

30. (Cancelled)

31. (Previously Presented) The receiver/decoder according to claim 20, wherein said downloading means is arranged to download a second loader included in the software included in said bitstream for downloading one of the first-mentioned loader and the software.

32. - 33. (Cancelled)

34. (Previously Presented) The transmission system according to claim 53, wherein said tables have respective different TID-extensions other than a predetermined TID-extension; said system further comprising means for generating a respective directory tables for the plurality of modules having the same TID, each directory table having that TID and said predetermined TID-extension, the directory table containing for each of the modules a name of that module and the respective TID-extension.

35. (Previously Presented) The transmission system according to claim 52, further comprising:

> means for generating a directory table having a predetermined table identification ("TID") and containing, for each of a plurality of version identifications of a receiver/decoder, a respective TID associated with that version identification.

36. (Previously Presented) The transmission system according to claim 52, further comprising means for including in each transmitted table a version identification therefor.

39. (Previously Presented) A signal including at least one loader for loading a software in native code into a receiver/decoder, and the software associated with the at least one loader, the at least one loader being divided into a plurality of modules and the software associated with the at least one loader being divided into a respective plurality of

modules.

43. (Previously Presented) The method according to claim 1, wherein the bitstream includes at least one data loader, said method further comprising:

  dividing the at least one data loader into a plurality of modules; and

  dividing the software into a respective plurality of modules, each plurality of the software modules being associated with a respective plurality of loader modules.

44. (Currently Amended) The method according to claim 43, further comprising:

  formatting the plurality of data loader modules as respective tables, the tables having the same respective table identification ("TID") and respective different table identification extensions ("TID-extensions")—at—a receiver/decoder; and

  formatting the plurality of the software modules as a respective table, the tables having the same respective TID as the tables of the loader modules associated therewith and respective different TID-extensions—at—the receiver/decoder.

45. (Previously Presented) The method according to claim 9, wherein said tables have respective different TID-extensions other than a predetermined TID-extension, and further comprising:

  generating a respective directory table for the plurality of modules having the same TID, the directory table having said predetermined TID-extension and the same TID, the directory table containing for the plurality of modules a name of a module and a respective TID-extension.

46. (Previously Presented) The method according to claim 45, further comprising:

  downloading one of the tables having the predetermined TID-extension so as to download a directory table;

  determining from the content of the directory table the TID-extensions of the

module tables having the same TID as the directory table; and

downloading the module tables having the same TID as that of the downloaded directory table and TID-extensions determined from the downloaded directory table.

47. (Previously Presented) The method according to claim 1, further comprising:

generating a directory table having a predetermined table identification ("TID") and containing, for a plurality of version identifications of a receiver/decoder, a respective TID associated with that version identification.

48. (Previously Presented) The method according to claim 13, further comprising:

downloading said directory table having the predetermined TID; and determining the version identification of the receiver/decoder, wherein downloading a directory table comprises downloading that one of the tables having a TID associated with a version number of the receiver/decoder and a predetermined TID-extension.

49. (Previously Presented) The method according to claim 45, further comprising:

including in a transmitted directory table a directory version identification therefor;

determining at the receiver/decoder whether the directory version identification of a currently transmitted directory table is more recent that the directory version identification of a previously downloaded directory table having the same TID as said currently transmitted table; and

aborting downloading the data if the currently transmitted directory table is not more recent.

50. (Previously Presented) The method according to claim 1, further comprising:

including in the bitstream a software version identification of the software;

determining, at the receiver/decoder, whether the software version identification

of received software is more recent than the software version identification of currently stored software; and

downloading the received software from the bitstream if the received software is more recent.

51. (Currently Amended) The method according to claim 1, further comprising:

transmitting a second loader included in said bitstream, ~~at a~~ to the receiver/decoder;

downloading the second loader, at the receiver/decoder; and

downloading the loader and the software using the second loader.

52. (Previously Presented) A transmission system comprising:

means for transmitting a bitstream including at least one loader for loading an application in native code into a receiver/decoder, and software in native code associated with the at least one loader; and

means for dividing the at least one loader into a plurality of modules and dividing the software associated with the at least one loader into a respective plurality of modules for transmittal by said transmitting means.

53. (Previously Presented) The transmission system according to claim 52, further comprising:

means for formatting each of the modules of the at least one loader as a respective table, the table of the at least one loader having the same respective table identification ("TID") and respective different table identification extensions ("TID-extensions"); and

means for formatting each of the modules of the software associated with the at

least one loader as a respective table, the table of the modules of data

having the same respective TID as the tables of the loader modules

associated therewith and respective different TID-extensions.

54. (New) A method for updating resident software to a receiver/decoder, comprising:

downloading an updated resident software using a resident loader into the receiver/decoder;

updating the resident software in the receiver/decoder, wherein updating the resident software comprises replacing resident software with the updated resident software; and

deleting the resident loader from the receiver/decoder.

55. (New) A receiver/decoder, comprising:

resident software executing on the receiver/decoder,

a bootstrap loader configured to download a loader from a bit stream; and

a memory configured to store the loader and the resident software,

wherein the loader is configured to download an updated version of the resident software, and

wherein the receiver/decoder is configured to update the resident software using the updated version of the resident software;

wherein the receiver/decoded is configured to delete the loader once the resident software is updated.